ซอฟต์แวร์
อาจหาญ สัตยารักษ์

เทคนิคการป้อน ข้อความ ในกราฟิกโหมด

มีเทคนิคป้อนข้อความ ลงในโหมดกราฟิก ของภาษาปาสคาลได้แล้ว

มื่อผู้เขียนได้เขียนถึงการสร้างอักขระกราฟิกมา 2 ครั้งแล้ว คือในฉบับที่ 50 และฉบับที่ 52 ก็มีสิ่งต่อเนื่องจากเรื่องนั้นเกิดขึ้นนั่นคือ กราฟิกของปาสคาลไม่มีส่วนการป้อนข้อความในโหมดกราฟิก เมื่อเราจะป้อนข้อความใด ๆ จึงเกิดปัญหาขึ้น

เพื่อให้ผู้อ่านมีโปรแกรมภาษาไทยที่สมบูรณ์แบบใช้ กล่าวคือมีทั้งส่วนแสดงข้อความ และส่วนของการป้อนข้อความ ซึ่งนับว่าจะได้ชุดภาษาไทยกราฟิกที่สมบูรณ์แบบพร้อมจะนำไปพัฒนางานของตนเอง เช่น เขียน CAI ช่วยการเรียนการสอนหรือพัฒนาไปเป็นโปรแกรมพวกเวิร์ดภาษาไทยก็จะสามารถทำได้

วันนี้เราจะมาพูดถึงการป้อนข้อความในกราฟิกโหมดกัน

เลือกรูปแบบภาษาไทย

เนื่องจากผู้เขียนเห็นว่าอักขระกราฟิกที่เขียนถึงในฉบับที่ 52 นั้นมีความสมบูรณ์กว่าที่กล่าวถึงในฉบับที่ 50 หรืออีกประการหนึ่งผู้อ่านที่ชอบนั่งคีย์โปรแกรมเอง (ซึ่งความจริงจะเกิดประโยชน์ต่อการศึกษามากกว่าแบบใช้บริการซอฟต์เมล์) จะได้สามารถเขียนขึ้นใช้เองได้ เพราะไฟล์ตัวอักษรของราชวิถี WORD PC นั้นคงหาได้ไม่ยากนัก อีกทั้งสามารถแสดงรูปแบบของตัวอักษรได้มากแบบกว่า จึงขอเลือกใช้ภาษาไทยกราฟิกของฉบับนี้มาเป็นต้นแบบของการป้อนข้อความที่จะกล่าวถึงต่อไป

ผู้อ่านที่ป้อนไฟล์ของฉบับที่ 52 ไว้แล้วก็คงเหนื่อยน้อยลง ส่วนผู้ที่ยังไม่ได้ป้อนไว้และไม่อยากเหนื่อยก็ติดต่อใช้บริการซอฟต์เมล์ก็จะได้ครบถ้วนเหมือนกันแต่ดีกว่าของฉบับที่ 52 (มีไฟล์เดียวชื่อ InputGr.pas)

โดยผู้เขียนจะเลือกใช้ตัวอักษรขนาดเล็กที่สุด คือ 8×8 จุด มาป้อนข้อความ ส่วนใครต้องการใช้ตัวอักษรขนาดอื่นก็ให้ดัดแปลงเอาเองได้ เพราะผู้เขียนจะอธิบายหลักการให้ทราบเป็นขั้น ๆ ไป

ดัดแปลงไฟล์ใช้งานบางไฟล์

เพื่อให้การใช้งานป้อนข้อความสมบูรณ์ขึ้น กับเพื่อให้การอ่านไฟล์เก็บรูปแบบตัวอักษรไวขึ้น ผู้เขียนได้แก้ไขโปรแกรมในรูปที่ 4 ของฉบับที่ 52 (หน้า 281) มาใช้โปรแกรมในรูปที่ 1

```
uses crt,graph,dos;
const Bit :array[0..7] of byte=(1,2,4,8,16,32,64,128);
Type Font=array [1..44] of byte;
var  FontFile:File;     k,FontCode,Mode:byte;
     Font1,Font2:array[0..223] of Font;
     FontDisp:Font;          ch:char;
     putck,ASCII:byte;
     InputSt:string; kk,x:word;

PROCEDURE OpenGraph;
var
      GrDriver,GrMode :integer;
begin
   GrDriver :=0; GrMode:=0;
  InitGraph(GrDriver, GrMode, '');
end; { OpenGraph  }

PROCEDURE SaveFont;
var   size:word;
 BEGIN
     clrscr; textcolor(0); TextBackground(7);
     writeln('                                                ');
     writeln(' Create Font for Graphics with WORD PC ');
     writeln('  By Arthan Sattayarak  Lampang 52000  ');
     writeln('                                                ');
     textcolor(3); TextBackground(0);  Gotoxy(1,7);
     writeln('Loading FONT1.NLQ');
       Assign(FontFile,'font1.nlq');
          Reset(FontFile,1);
          size:=Filesize(FontFile);
          blockRead(FontFile,Font1,size);
       Close(FontFile);
     writeln('Loading FONT2.NLQ');
       Assign(FontFile,'font2.nlq');
          Reset(FontFile,1);
          size:=Filesize(FontFile);
          blockRead(FontFile,Font2,size);
       Close(FontFile); FontCode:=0; Mode:=0;
end; { SaveFont }
```

รูปที่ 1 โปรแกรมส่วนกำหนดตัวแปร และ การอ่านไฟล์ที่พัฒนาจากของเดิม (ฉบับที่ 52)

ของฉบับนี้แทน โดยมีการให้ตัวแปรเพิ่มขึ้นคือ

InputSt เพื่อเก็บข้อความที่ได้ทำการป้อนเมื่อเรียกใช้ชุดคำสั่ง InputStGr ที่จะกล่าวถึงต่อไป

ASCII โดยตัวแปร ASCII เป็นตัวแปรที่ย้ายจากชุดคำสั่ง PrintFont (รูปที่ 9 ฉบับที่ 52) และให้ตัดตัวแปร ASCII นี้ออกไปจากชุดคำสั่ง PrintFont ดังกล่าวด้วย เนื่องจากตัวแปร ASCII นี้ผู้เขียนจะนำมาใช้เป็นเงื่อนไขในการแสดงอักขระกราฟิกแบบกลับขาวดำ

Putck เป็นตัวแปรที่กำหนดขึ้นมาใหม่ เพื่อจะนำไปเป็นเงื่อนไขในการแสดงตัวอักขระ 3 แบบ คือ แสดงแบบตัวนูน (ShaDow), แสดงแบบตัวโปร่ง (Outline) และแบบกลับขาวดำ (Inverse)

ส่วน kk,x กำหนดเพื่อใช้ในการป้อนข้อความ

ส่วนการอ่านไฟล์นั้นได้เปลี่ยนมาใช้คำสั่ง Blockread แทนของเดิมที่อ่านทีละข้อมูล ซึ่งการอ่านแบบนี้จะใช้เวลาน้อยกว่า ลองศึกษารูปแบบการอ่านไฟล์ทั้งสองเปรียบเทียบกันด้วยเพื่อจะได้เข้าใจในเรื่องนี้ดีขึ้น

พัฒนารูปแบบการแสดงอักขระ

ในฉบับที่ 52 นั้นผู้เขียนได้บอกแล้วว่าจะสามารถนำไปสร้างตัวอักขระแบบอื่นได้ด้วย ซึ่งคิดว่าผู้อ่านบางท่านคงนำไปพัฒนาได้อีกหลายรูปแบบ

ครั้งนี้ขอเสนอรูปแบบเพิ่มเติมอีก 3 รูปแบบ โดยอาศัยตัวแปร Putck มาควบคุม ดังนี้

ตัวแปร PutCk=1 จะแสดงผลแบบปกติ เหมือนที่เห็นในฉบับที่ 52

ตัวแปร PutCk=0 จะแสดงผลแบบตัวโปร่ง

ตัวแปร PutCk=2 จะแสดงผลแบบตัวนูน

ตัวแปร PutCk=3 จะแสดงผลแบบกลับขาวดำ

โดยเสนอการแก้ไขมาให้เป็นตัวอย่างเพียงชุดเดียวเท่านั้น คืออักขระขนาด 16×16 จุด พร้อมกับขอยกเลิกชุดคำสั่ง Dot 16×16 (รูปที่ 5 หน้า 282 ฉบับที่ 52) มาใช้ชุดคำสั่งที่เขียนขึ้นใหม่นี้แทน (คือชุดคำสั่งชื่อเดิม ในรูปที่ 2 ของฉบับนี้) ส่วนชุดคำสั่งที่ยังคงเดิมก็คือ ชุดคำสั่งในรูปที่ 6-8 (ฉบับที่ 52) นั้นก็ให้นำมาใช้งานร่วมกันด้วย

ลองศึกษาการทำงานของชุดคำสั่งนี้ดู ว่ามีการแสดงอักขระแบบที่กล่าวถึงได้อย่างไร

หวังว่าคงเข้าใจการทำงานของชุดคำสั่ง Dot 16×16 แล้วนะ

การทำตัวอักขระแบบกลับดำขาวนั้น (ตัวแปร putck=3) ทำได้ดังนี้

1. ตรวจตัวแปร ASCII ถ้าน้อยกว่า 215 (คือต่ำกว่าสระ อุ ลงมา) ก็ให้สร้างกรอบขาวด้วยคำสั่ง bar แต่เนื่องจากภาษาไทยและภาษาอังกฤษ

```
PROCEDURE Dot16x16(xx,Yy:integer);
const Ypix:array[0..7] of byte=(1,3,5,7,9,11,13,15);
var Byte_No,Dot:integer; OneByte,Check,TimeCk:byte;
    Put_ck,x,y,xxck:integer;

begin
   Byte_No:=1;  Check:=0; Put_ck:=27;  x:=0; y:=0;
   Timeck:=0;   xxck:=xx;
   if putck<>1 then Put_ck:=1; dec(xx); dec(yy);
   if putck=3 then begin  setfillstyle(1,1); Put_ck:=0;
           case ASCII of
            0..127: Bar(xx,yy-30,xx+22,yy+8);
            160..214: Bar(xx,yy-28,xx+22,yy+10);
           end;     end;
Repeat
 Repeat
  OneByte := FontDisp[Byte_No];
  for Dot := 0 to 7 do
      if OneByte and Bit[Dot]<>0 then
           putpixel(xx+x,yy-YPix[Dot]+y+Check mod 2,put_ck);
    Check:=Check+1;  xx:=xx+1-Check mod 2;
  if Check mod 2=1 then Byte_No:=Byte_No+22;
  if check mod 2=0 then Byte_No:=Byte_No-21;
 until Byte_No>=44;  Byte_no:=1; check:=0;
  if putck=0 then  begin  Inc(Timeck); inc(x);   xx:=xxck;
        if Timeck=3 then begin x:=0; Inc(y,2); end;
        if timeck=6 then begin x:=1; y:=1; put_ck:=0; end;
        if Timeck=7 then Put_ck:=27;
                 end;
  if putck=2 then  begin Inc(Timeck); Dec(x,2); Dec(y,2); xx:=xxck;
        if Timeck=3 then Put_ck:=27;
                 end;
  if putck=3 then  Put_ck:=27;
 Until Put_ck=27;
End;
```

รูปที่ 2 ชุดคำสั่ง Dot 16x16 ที่พัฒนาการแสดงรูปแบบเพิ่มอีก 3 แบบ

เขียนที่ระดับ x,y ต่างกัน จึงต้องสร้างกรอบที่ตำแหน่งไม่เท่ากันด้วย

2. ให้ลงจุดด้วยสี 0 (Put_ck =0) โดยที่ Put_ck เป็นตัวแปรที่กำหนดให้ลงสีเป็น 0 หรือ 1 แค่นี้ก็จะได้อักขระแบบกลับขาวดำ

การทำตัวอักขระแบบตัวนูน (ตัวแปร putck=2) ทำได้ดังนี้ ทำการวาดภาพอักขระตัวละ 3 ครั้ง โดยทำการขยับค่า x, y ไปครั้งละ 2 จุดเสมอ

การทำอักขระแบบตัวโปร่ง (ตัวแปร putck=0) ทำได้ดังนี้

1. กำหนดตัวแปร put_ck=1 เพื่อลงภาพอักขระรวม 6 ครั้ง โดยเรียงกันเป็นตัวหนา 3 จุดติดกัน

2. กำหนดตัวแปร put_ck=0 แล้วลงภาพ(ความจริงลบภาพ) ครั้งที่ 7 ที่ตรงกลางของภาพที่เขียนมาตามข้อหนึ่ง

ทำอย่างนี้ก็จะได้ภาพอักขระตัวโปร่งมาใช้แล้ว

ตัวอย่างทั้ง 3 แบบปรากฏในรูปที่ 5 และจากคำสั่งนี้เราก็น่าจะสร้างอักขระแบบตัวหนาได้ด้วย

สำหรับที่ท่านต้องการแสดงผลแบบนี้ที่ตัวขนาดอื่นก็ขอให้แก้ไขโปรแกรมที่ 6-9 (ฉบับที่ 52) ตามแนวที่อธิบายมานี้ได้เช่นกัน

การป้อนข้อความ

เรามาเริ่มเรียนส่วนการป้อนข้อความกันต่อไป

ขนาดตัวอักขระกราฟิกที่เลือกมาใช้เป็นตัวอย่างคือตัวขนาด 8x8 จุด ซึ่งตัวแบบนี้เราได้กำหนดให้มีระยะห่างของแต่ละตัวอักขระไว้ 11 จุด ดังนั้นเราจึงได้ขอบเขตที่ควรสนใจดังนี้

สมมุติเราแสดงอักขระที่ตำแหน่ง kk,row (kk ตำแหน่งในแนวนอน Row ตำแหน่งในแนวตั้ง)

1. ตัวอักขระกลางทั้งหมด รวมสระที่อยู่ในระดับเดียวกัน จะมีกรอบขนาด 11x13 (เผื่อสำหรับ ฎ ฏ ฐ ญ ฤ) นั้นคืออยู่ในกรอบ ขนาด kk, row-10,kk+10,row+2 ซึ่งถ้าจะมีการลบอักขระกลุ่มนี้ก็ต้องลบที่บริเวณกรอบที่คำนวณไว้นี้ด้วยจึงจะลบได้หมดพอดี พวกนี้จะมีค่า ASCII ต่ำกว่า 215 ลงมา

2. สระล่างคือ สระ ุ ู จะมีขนาดประมาณ 11x8 อยู่ที่ส่วนล่างของตัวอักขระในข้อ 1 หรือมีขอบเขตขนาด kk,row+2,kk+10,row+9 แต่ถ้าการลบอักขระกลุ่มนี้เราจะไม่มีการเลื่อนเคอร์เซอร์ ดังนั้นขอบเขตการลบจึงต้องขยับไปทางซ้ายอีก 11 จุด ซึ่งจะใช้ขอบเขต kk-10,row+2,kk,row+9 แทน รหัส ASCII ส่วนนี้คือ 215 และ 216 ตามลำดับ

3. สระบน คือ ิ ี ึ ื ั ่ ้ ๊ ๋ จะอยู่ข้างบนในขอบเขต kk, row-15,kk+10,row-9 ถ้าจะลบก็ต้องขยับไปลบทางซ้าย 11 จุดเช่นเดียวกัน ส่วนนี้กำหนดไว้สูงเพราะต้องเผื่อไว้ลบก์ ที่จะมาแสดง ฌ ส่วนนี้ด้วย (ค่า ASCII ของส่วนนี้อยู่ระหว่าง 217 ถึง 223)

4. วรรณยุกต์ คือ ็ ่ ้ ๊ ๋ พวกนี้เรากำหนดไว้ในฉบับก่อนให้แสดงได้ 2 ระดับ คือในตำแหน่งเดียวกับกลุ่มที่ 3 ถ้าไม่มีการป้อนอักขระกลุ่มที่ 3 และอยู่สูงขึ้นกว่ากลุ่มที่ 3 ในกรณีที่มีอักขระกลุ่มที่ 3 เก็บไว้ใน

ตัวแปร Inputst ล่วงหน้า 1 อักขระ คือแสดงในขอบเขต kk,row-18,kk+10, row-12 (ส่วนนี้มีค่า ASCII ตั้งแต่ 224-228)

5. ไม่ต้องตรวจสอบสระผสม เพราะว่าตอนป้อนข้อความนั้น ข้อความจะเรียงกันแบบธรรมดา ไม่ใช่นำไปผสมกัน

6. ขนาดเคอร์เซอร์ที่ใช้ก็คือขนาด 11×12 จุด โดยจะแสดงตัวอักษร T และ E ตรงกลาง เพื่อให้รู้ว่าอยู่ในโหมดภาษาไทยหรือภาษาอังกฤษ

ถ้าผู้อ่านจะใช้ตัวอักษรขนาดอื่นก็ต้องกำหนดรูปแบบของเคอร์เซอร์, การลบข้อความ จากแนวคิดนี้ให้เหมาะสมกับขนาดของอักขระด้วย

ถึงตรงนี้อยากจะให้ผู้อ่านกลับไปศึกษาโปรแกรมในรูปที่ 3 ดูก่อน เพื่อจะได้ศึกษาได้เองก่อนการอ่านคำอธิบายที่จะเขียนถึงต่อไป

ตรวจการป้อนภาษาไทย

คราวนี้ก็มาถึงปัญหาว่า เราจะรู้ได้อย่างไรว่าเป็นภาษาไทยหรือภาษาอังกฤษ และถ้าเป็นภาษาไทยเราจะรู้ได้อย่างไรว่า กดแป้น ก, ข หรือ ค

ลองดูการวางแป้นพิมพ์ภาษาไทยที่เราใช้กันอยู่จะเป็นอย่างนี้ (ขอยกตัวอย่างเพียงบางส่วนเท่านั้น) ดังตารางที่ 1

นั่นคือเมื่อกดแป้น a (ไม่กด Shift) จะได้ค่า ASCII=97 เมื่ออยู่ภาษาไทยจะต้องเปลี่ยนรหัสนี้ไปเป็นตัว ฟ ซึ่งมี ASCII=189 แต่เมื่อกด A (กด Shift ด้วย) จะได้ค่า ASCII=65 แต่ภาษาไทยเป็นตัว ฤ รหัส ASCII=194 หรือตัวอักขระอื่นก็เป็นแบบเดียวกัน เช่น

```
PROCEDURE clscr(x1,y1,x2,y2:integer);
begin
    setviewport(x1,y1,x2,y2,clipon); clearviewport;
    setviewport(0,0,GetMaxX,GetMaxY,clipon);
end;

PROCEDURE ssound(s,d:word);
begin sound(s); delay(d); nosound;
end;

PROCEDURE InputStGr(Column,Row,Count:byte);
 CONST
ThaiASCII:array[32..126] of byte=(32,49,46,51,52,53,55,165,57,48,56,
      43,191,162,211,187,166,195,47,45,190,180,215,219,163,179,169,196,
      176,168,201,165,50,194,44,167,173,172,210,170,223,177,227,198,197,
      63,228,214,171,54,175,164,182,226,203,34,41,222,40,184,92,195,216,
      95,96,189,217,209,161,207,178,208,225,193,224,206,199,181,220,183,
      192,213,188,200,204,218,202,212,185,221,186,174,124,125,126);
cursorst:array[0..1] of char=('E','T');
var xpix,y,kk:word;  tmode,Y_pix:byte; stck:string[5];

begin  tmode:=1;  PrintFont(740,10,#11#3' ');
       kk:=Column; InputSt:=''; y:=0;  x:=0; ch:=#0;
      repeat
        if y=0 then begin y:=4; bar(kk,row-10,kk+10,row+2);
           setcolor(0); outtextxy(kk+2,row-7,cursorSt[Tmode]);
           setcolor(1);  end;
   if keypressed then begin  ch:=readkey; y:=0;
      clscr(kk,Row-10,kk+10,Row+2);
      if ch=#27 then begin inc(Tmode); Tmode:=Tmode mod 2; end;
      if (tmode=1) and (ch>#31) then ch:=chr(ThaiAscII[ord(ch)]);
      if (inputst[length(inputst)] in [#217..#223]) and
         (ch in [#217..#223]) then      begin
           ch:=#0;  ssound(90,300);  end;
      if (inputst[length(inputst)] in [#215,#216,#172..#174,#194]) and
         (ch in [#215,#216]) then       begin
           ch:=#0;  ssound(90,300);  end;
      if (inputst[length(inputst)] in [#32..#127,#204..#206,#208..#214])
         (ch in [#215..#228]) then      begin
           ch:=#0;  ssound(90,300);  end;
      if (inputst[length(inputst)] in [#224..#228]) and
         (ch in [#224..#228]) then      begin
           ch:=#0;  ssound(90,300);  end;
    if (ch in [#32..#255]) and (x<Count) then begin stck:='
          Inc(x); stck[1]:=ch; xpix:=0;
      if ch=#207 then                                    begin
        stck:=inputst[length(inputst)]+ch;  xpix:=11;  end;
      if ch in [#224..#228] then      begin
        if inputst[length(inputst)] in [#217..#223] then  begin
          stck:=inputst[length(inputst)-1]+inputst[length(inputst)]+ch;
                                          xpix:=11;       end;
      if inputst[length(inputst)] =#207 then        begin
        stck:=inputst[length(inputst)-1]+inputst[length(inputst)]+ch+'
                                        xpix:=22;  end;
                                          end;
        InputSt:=InputSt+ch;  PrintFont(kk-xpix,Row,Stck);
        Inc(kk,11); if ch in [#215..#228] then dec(kk,11);  end;

     if (ch=#8) and (x>0) then begin
        if inputst[length(inputst)] in [#32..#214] then   begin
             clscr(kk-10,row-15,kk,row+9); Dec(kk,11); end;
        if inputst[length(inputst)] in [#215,#216] then
                     clscr(kk-11,row+2,kk,row+9);
        if inputst[length(inputst)] in [#207,#217..#223] then
                    clscr(kk-11,row-15,kk,row-9);
        if inputst[length(inputst)] in [#224..#228] then    begin
          if inputst[length(inputst)-1] in [#217..#223] then
                     clscr(kk-11,row-18,kk,row-12);
          if inputst[length(inputst)-1] in [#32..#206,#208..#216] then
                     clscr(kk-11,row-15,kk,row-9);
          if inputst[length(inputst)-1] =#207 then
                     clscr(kk-22,row-15,kk-10,row-12);   end;
        InputSt:=Copy(InputSt,1,Length(InputSt)-1);  Dec(x);
                                  end;
                           end;
       until ch=#13;
end;
```

รูปที่ 3 โปรแกรมสำหรับป้อนข้อความโดยใช้ร่วมกับโปรแกรมในรูปที่ 1,2 และฉบับที่ 52

ตารางที่ 1

ตำแหน่งปกติ	A	S	D	F	G	H
ไม่กด Shift	ฟ	ห	ก	ด	เ	้
กด Shift	ฤ	ฆ	ฎ	โ	ฌ	็

อักขระที่เครื่องรับรู้เมื่อกดแป้น						
a	s	d	f	g	h	
A	S	D	F	G	H	

(แป้น Caps Lock อยู่ในตำแหน่ง off)

กด f (ASCII = 102) จะได้ ด (ASCII = 178) กด F (ASCII = 70) จะได้ โ (ASCII = 210)

กด d (ASCII = 100) จะได้ ก (ASCII = 161) กด D (ASCII = 68) จะได้ ฎ (ASCII = 173)

โดยลักษณะนี้เราจึงสร้างตารางขึ้นมาตารางหนึ่งเพื่อเปลี่ยนรหัส ASCII ที่อ่านได้จากการกดแป้นไปเป็นรหัสภาษาไทย สมมติผู้เขียนให้ชื่อตารางนี้ว่า ThaiASCII โดยเป็นตารางแบบมิติตั้งแต่ 32-126 ก็จะให้รหัสของ ฟ และ ฤ ดังนี้ ThaiASCII[65] = 194 และ ThaiASCII[97] = 189 โดยมีตัวแปรอีกตัวหนึ่งเป็นตัวตรวจสอบว่าอยู่ในภาษาไทยหรือภาษาอังกฤษ (สมมุติชื่อตัวแปรว่า TMode โดยถ้า TMode = 1 ก็จะเป็นภาษาไทย)

การสร้างตารางจริง ๆ ในปาสคาลเราก็กำหนดเป็นค่าคงที่ (const) ได้เลย ขอให้ดูจากโปรแกรมในรูปที่ 3 คือ ThaiASCII:array[32..126] of byte =(32,49,....124,124,126); ซึ่งบางรหัสได้กำหนดไว้ล่วงหน้าเพื่อจะใช้ประโยชน์ในภายหลัง เช่น อาจนำไปสร้างเลขไทย ฯลฯ เป็นต้น

ถ้าลองนับดูจะเห็นว่าลำดับที่ 65 คือ 194, ลำดับที่ 97 คือ 189 ถ้าอยากได้ตำแหน่งของแป้นพิมพ์เป็นอย่างอื่นก็กำหนดรหัสใหม่ได้ตามที่ท่านต้องการ

สร้างเคอร์เซอร์กราฟิกและแป้นควบคุมไทย/อังกฤษ

เพื่อให้รู้ว่ากำลังอยู่ในการป้อนภาษาไทยหรือภาษาอังกฤษ เราก็สร้างเคอร์เซอร์กราฟิกขึ้นมาช่วย โดยสร้างแบบง่าย ๆ คือ วาดกรอบสี่เหลี่ยมขนาด kk,row-10,kk + 10,row + 2 ขึ้นมาแล้วเขียนอักษร T หรือ E ด้วยสีดำ setcolor (0) ที่ตรงกลางกรอบนี้ก็จะได้เคอร์เซอร์กราฟิกตามต้องการ โดยตัวอักษรทั้งสองนี้ได้กำหนดไว้ในค่าคงที่ 2 มิติที่ชื่อ cursorst

ตามโปรแกรมในรูปที่ 3 นั้นเป็นเคอร์เซอร์แบบไม่กะพริบ ถ้าอยากได้เคอร์เซอร์กระพริบก็เขียนคำสั่งเพิ่มได้ โดยคิดว่าเพิ่มอีก 2 บรรทัดก็จะกลายเป็นเคอร์เซอร์กระพริบได้แล้ว

เมื่อมีการป้อนข้อความก็ขยับค่า kk ไปทางขวาอีก 11 จุด ด้วยคำสั่ง Inc(kk,11)

เมื่อมีการลบข้อความก็ขยับค่า kk ไปทางซ้าย 11 จุด ด้วยคำสั่ง Dec(kk,11)

โดยตรวจสอบด้วยว่าเป็นสระหรือวรรณยุกต์หรือไม่ เพราะไม่จำเป็นต้องขยับในบางครั้งด้วย

ตอนเข้ามาในชุดคำสั่ง Input-StGr นี้จะกำหนดให้อยู่ในภาษาไทยเลย TMode = 1; โดยการกำหนดค่า Column, Row เป็นตำแหน่งเริ่มต้นของการป้อน count เป็นจำนวนอักขระสูงสุดที่ต้องการให้ป้อน

ส่วนการเปลี่ยนโหมด ไทย/อังกฤษนั้นใช้แป้น ESC (#27) หากท่านต้องการเป็นแป้นอื่นก็ให้เปลี่ยนค่า #27 นี้ไปเป็นของแป้นที่ท่านต้องการได้ แต่ต้องคำนึงว่าแป้นที่ใช้นี้ต้องไม่ใช่แป้นที่ใช้กดอักขระภาษาไทยด้วย โดยทุกครั้งที่มีการกดแป้น ESC นี้ค่า TMode จะถูกเปลี่ยนไปมาระหว่าง 0 และ 1 จากการเพิ่มค่าในคำสั่ง Inc (TMode); Tmode: = TMode mod 2;

การตรวจแป้นเพื่อป้องกันการพิมพ์ซ้ำ

การป้อนข้อความภาษาไทยนั้นต่างจากภาษาอังกฤษตรงที่ของเราจัด 3 ระดับ จึงต้องยุ่งยากต่อในการตรวจสอบการกดแป้นเป็นธรรมดา เช่น ไม่ยอมให้มีการป้อนสระบน คือ ◌ิ ◌ี ◌ึ ◌ื ◌ั ◌็ ซ้ำเมื่อได้ป้อนตัวอื่นไปแล้ว หรือคำว่า ขั้น ก็อยากให้คนป้อนได้เพียงแบบเดียว คือ ข ◌ั ◌้ น โดยไม่ยอมให้ป้อน ข ◌ั ◌้ น คำว่า กุ้ง ก็อยากให้ป้อนเรียงลำดับเป็น ก ◌ุ ◌้ ง เป็นต้น

การทำแบบนี้มีข้อดีคือ เมื่อเราเขียนไปเก็บในเท็กซ์ไฟล์ ก็จะสามารถค้นหาได้โดยง่าย ไม่ต้องไปสับเปลี่ยนรหัสใด ๆ อีก

ส่วนถัดมาจึงเป็นการตรวจเกี่ยวกับเรื่องนี้

โดยจะใช้การตรวจจากตัวสุดท้ายของตัวแปร InputSt ว่าเป็นรหัสพวกนี้หรือไม่ เช่น ถ้ากดแป้น (#216) ก็ตรวจว่าท้ายสุดของตัวแปร InputSt เป็น #215..#216 หรือไม่ ถ้าเป็นก็ให้ค่า ch:= #0; ซึ่งเป็นการยกเลิกการกดแป้นครั้งนี้ไป พร้อมกับเปล่งเสียงให้ทราบ (เรียกใช้ชุดคำสั่ง SSound; อีกที)

การตรวจสระ ฯลฯ นั้นก็ใช้แนวคิดเดียวกัน จากชุดคำสั่งนี้จะป้อน ต่ำ ต้องป้อนให้เป็นดังนี้ ต ำ ' คือป้อนตามการเขียนอักขระที่ถูกต้อง (ในเท็กซ์โหมดต้องป้อน ต ' ำ เท่านั้น)

การพิมพ์ข้อความให้ปรากฏบนจอภาพนั้นถ้าเป็นการกดแป้นสระ เราก็ส่งข้อความย้อนหลังไปแสดง ก็เพื่อให้ชุดคำสั่ง Dot 8×8 สามารถแสดงอักขระในตำแหน่งที่ถูกต้อง ตัวแปรที่ส่งไปแสดงบนจอภาพก็คือตัวแปรชื่อ StCk นั้นเอง เช่น ข้อความก่อนหน้านี้เป็น การท แล้วเราป้อน ี ก็กำหนดให้ตัวแปร StCk นี้ เป็น ที และเมื่อป้อน ' ต่อ ก็จะได้ค่า Stck เป็น ที่ ส่งไปให้แสดงผลบนจอภาพได้อย่างถูกต้อง

ทุกครั้งที่มีการป้อนความได้สมบูรณ์เราเพิ่มค่า x หนึ่งค่าเสมอ Inc(x) เพื่อใช้ตรวจสอบไม่ให้เกินกว่าจำนวนที่เรากำหนดไว้(ไม่มากไปกว่าตัวแปร count)

การตรวจสอบเมื่อมีการลบข้อความ

การป้อนข้อความผิดพลาดมักเกิดขึ้นเสมอ ดังนั้นเขียนส่วนป้อนข้อความโดยไม่ยอมให้ลบข้อความที่ป้อนผิดพลาด ก็คงจะทำให้โปรแกรมของเราไม่สมบูรณ์แน่

เราจึงต้องตรวจสอบการลบข้อความด้วยเช่นกัน

แป้นที่ใช้ก็คือแป้น back space (แป้นนี้มีค่า ASCII=8 หรือคือ #8 ในภาษาปาสคาล) นั่นเอง

ขอให้ดูคำสั่งการตรวจสอบนี้ซึ่งอยู่ด้านท้ายของชุดคำสั่ง InputStGr

โดยมีแนวคิดแบบการตรวจการป้อน คือ ตรวจที่ตัวสุดท้ายของ Input-St เพื่อจะได้ทำการลบจอภาพได้ตรงกับตำแหน่งของอักขระที่ต้องการลบ เราจึงต้องตรวจย้อนไปถึงตัวอักขระที่อยู่รองสุดท้ายด้วย เพราะในข้อความว่า ศักดิ์เราต้องตรวจว่า ์ แสดงที่ตรงไหน เป็นต้น การลบก็จะลบในขอบเขตที่ผู้เขียนได้กล่าวถึงในหัวข้อ การป้อนข้อความ เพียงแต่เราอาจต้องขยับไปทางซ้ายเพิ่มขึ้นเท่านั้น

หวังว่าคงศึกษาคำสั่งส่วนนี้ได้เอง

การป้อนข้อความจะจบลงเมื่อมีการกดแป้น Enter หากท่านต้องการให้ Count เป็นจำนวนคอลัมน์สูงสุดแทน ก็คงต้องแก้ไขโปรแกรมนี้อีกเล็กน้อย (คิดว่าไม่เกิน 2 แห่ง)

ข้อความที่ป้อนได้นี้จะกลับมายังส่วนกลางตามค่าตัวแปร Inputst ซึ่งจะนำไปเปลี่ยนเป็นตัวเลขได้ โดยการใช้ val เข้ามาช่วยหาค่า เช่น val (InputSt, kk,x); จะได้ตัวเลขในค่าตัวแปร kk ซึ่งตัวแปรนี้จะเป็นตัวเลขชนิดใดก็ได้ แต่ตัวแปร x ต้องเป็นตัวแปรชนิด Integer เท่านั้น

รูปที่ 4 คือส่วนโปรแกรมหลักในการแสดงข้อความ 3 แบบที่พัฒนาใหม่ และตัวอย่างของการเรียกใช้ชุดคำสั่ง InputStGr โดยผู้เขียนได้พิมพ์ภาพหน้าจอมาเป็นตัวอย่างดังรูปที่ 5 โดยข้อความ 4 บรรทัดแรกนั้นกำหนดให้แสดงผล และข้อความถัดมาสองบรรทัดสุดท้ายเกิดจากการป้อนข้อความ

```
{ ======================== InputGr.pas ========================}
 begin
   SaveFont;   opengraph;  putck:=1;
   Rectangle(0,0,719,347); Rectangle(2,2,717,345);
   putck:=2;
   PrintFont(60,35,#11+#1+'การแสดงอักขระแบบตัวบูน(Shadow)');
   Putck:=0;
   PrintFont(45,70,#12#1'อักขระกราฟิกแบบตัวโปร่ง(Outline)');
   Putck:=3;
   PrintFont(30,120,#12#1' การแสดงอักขระกราฟิก'#11'แบบกลับขาวดำ ');
   Putck:=0;
   PrintFont(16,160,#11#1'โปรแกรมป้อนข้อความภาษาไทย&English'); Putck:=1;
   InputStGr(50,210,120);
   InputStGr(10,230,120);
      ch:=readkey;
  closegraph;
 end.
```

รูปที่ 4 โปรแกรมหลักส่วนการแสดงข้อความ 3 แบบและส่วนการป้อนข้อความ

การแสดงอักขระแบบตัวทึบ(Shadow)
อักขระกราฟิกแบบตัวโปร่ง(Outline)
การแสดงอักขระกราฟิกแบบกลับขาวดำ
โปรแกรมป้อนข้อความภาษาไทย&English

การป้อนข้อความในโหมดกราฟิก ใช้แป้น ESC เข้า-ออก ไทย/Eng
มีของแถม คือ การทำอักษรแบบมีเงา, ตัวโปร่ง, ตัวกลับขาวดำ เพิ่มเติมให้จากเดิม

รูปที่ 5 ภาพที่ได้จากการแสดงผล 3 แบบ และข้อความที่ป้อน

ผลสุดท้ายผู้อ่านก็มีภาษาไทยกราฟิกดี ๆ ไว้ใช้งานได้อย่างครบถ้วนแล้ว คือ แสดงอักขระได้หลายแบบ สามารถป้อนข้อความบนกราฟิกได้ จึงหวังว่าท่านคงจะพัฒนาโปรแกรมด้านกราฟิกดี ๆ ขึ้นมาใช้กันมากยิ่งขึ้น

โชคดีทุกท่านครับ พบกันใหม่.. ฉบับต่อไป

รับสมัครด่วน

1. SALES MANAGER
2. SENIOR SALES
3. SALES

คุณสมบัติ
1. จบปริญญาตรี
2. การขายเครื่องคอมพิวเตอร์มีประสบการณ์ ไม่น้อยกว่า 1 ปี โดยเฉพาะระดับ MINI MAINFRAME จะพิจารณาเป็นพิเศษ
3. มีรถยนต์ส่วนตัว
4. อายุไม่เกิน 35 ปี

สนใจติดต่อ
ORACLE SYSTEMS (THAILAND) CO.,LTD.
LEVEL 23 C.P. TOWER
313 SILOM ROAD BANGKOK 10500
TEL: 231-0600

บริษัท ซอฟแวร์ซิสเต็มเทคโนโลยี จำกัด

บริษัทฯ ผู้จำหน่ายคอมพิวเตอร์และอุปกรณ์คอมพิวเตอร์ ต้องการรับสมัครพนักงานเพิ่มในตำแหน่งฝ่ายอบรมและบริการลูกค้า มีคุณสมบัติดังต่อไปนี้

- วุฒิ ปวส., ปริญญาตรี ด้าน COMPUTER ขึ้นไป
- มีความสามารถในการใช้ WORD IRC, DBASE, LOTUS ได้เป็นอย่างดี
- มีประสบการณ์ในการสอนคอมพิวเตอร์อย่างน้อย 1 ปี
- ประชาสัมพันธ์ 1 ตำแหน่ง
- หญิง วุฒิการศึกษา ปวช.

สนใจสมัครได้ที่

บริษัท ซอฟท์แวร์วิสเต็มเทคโนโลยี จำกัด
2310/6 ตรงข้ามประตูน้ำ ม.เกษตร ถ.พหลโยธิน
บางเขน กรุงเทพฯ 10900 โทร. 579-4989